Software Requirements Specification

for

KeePass Password Safe

Requirements for Version 1.10

Prepared by Elia Kouzari

Software Engineering, Aristotle University Thessaloniki

17-February-2008

Table of Contents

1. Introduction

1.1 Purpose

This document includes software requirements for KeePass Password Safe, release number 1.10. KeePass Password Safe is an OSI Certified Open Source Software distributed under the terms of the GNU General Public License Version 2 or under. The system gives resolution to memorizing passwords problem. Its purpose is to keep all of the user's passwords, data, email accounts, usernames and URLs stored in a very secure, encrypted database, protected by a Master Password. The system is very small so it can be easily transferred from one computer to another. It provides several functionalities on the already encrypted data and the new ones to be inserted. The database produced, is protected by a Master Password only known by its inventor with no backup if lost.

1.2 Document Conventions

- When writing this document it was inherited that all requirements have the same priority.
- First there is presented an overall view about KeePass and then all features and functions are analyzed in detail.

1.3 Intended Audience and Reading Suggestions

This requirement document contains general information about KeePass, main classes and use cases, functions, features and special technologies. It describes in detail all that KeePass needs to work properly and with safety.

The rest of the document is divided into chapters for better understanding.

- In chapter 2 an overall description of KeePass is provided. First product perspective is presented with product features and main functions. Then follow user classes and characteristics, operating environments that KeePass supports as well as design and implementation constraints. After all that user documentation is presented and will provide you with more details about each feature's technology.
- In chapter 3 most important features are presented with detailed description, use cases and requirements.
- In chapter 4 user and communication interfaces are described.
- In chapter 5 requirements about safety and performance are presented.

This document is intended for

Developers: in order to be sure they are developing the right project that fulfills requirements provided in this document.
Testers: in order to have an exact list of the features and functions that have to respond according to requirements and provided diagrams.
Users: in order to get familiar with the idea of the project and suggest other features that would make it even more functional.
Documentation writers: to know what features and in what way they have to explain. What security technologies are required, how the system will response in each user's action etc.
Advanced end users, end users/desktop and system administrators: in order to know exactly what they have to expect from the system, right inputs and outputs and response in error situations.

1.4 Project Scope

KeePass Password Safe is a small system that can be easily transferred from computer to computer by a simple USB stick. Its purpose is to solve a problem that really bothers many people today when they have to choose from memorizing a lot of passwords to be secure or to use every time the same one so they won't forget it but risk be found out by others. So it provides you a very secure, encrypted database where you can keep inside all your passwords, usernames, email accounts, URLs, notes without any risk for others to find them. That is because KeePass Password Safe can lock every database with only one Master Password and/or key file. There are no duplicates, anywhere in your computer, of this Master Password and/or key file so in case of lost database cannot be opened by anyone. Not even by you and that is because there is no recovery password or back door.

KeePass Password Safe beside security also provides you with several functionalities in order to keep your database organized and up to date. Those are analyzed in the following pages.

More about KeePass you can find out at http://keepass.info/

1.5 References

More about KeePass can be found at

- *http://sourceforge.net/projects/keepass/*

  In this website you can find out more about the project and discuss any questions in the forums. You can go back and look at previous releases, code and problems that have been solved. There you can also find information about the developers as well as the project's main characteristics such as programming language and algorithms

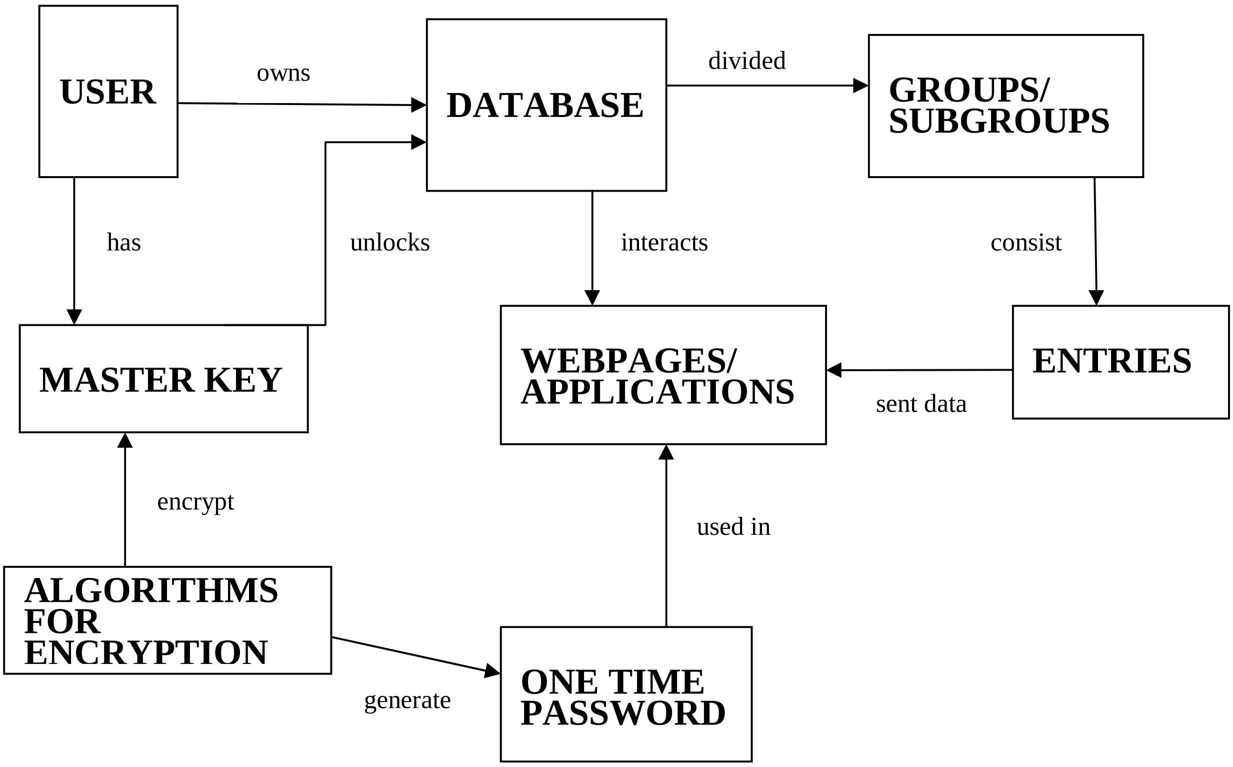
- *http://keepass.info/*

  This is project's official website where you can find links to all above and also find features available for downloading such as language translations and plug-ins.

2. Overall Description

2.1 Product Perspective

KeePass consists of a database which contains data for one or more users. Each user’s data are divided into groups and subgroups so that they are organized in a form that serves right the user. Every user has a unique Master Key which can be simple or composite and its combination opens uniquely the database. If lost there is no recovery. Groups and subgroups contain entries with usernames, passwords URLs etc that can be sent or copied to websites, application and accounts. There is also the ability for a onetime key creation to be used once in a transaction without the risk of reused by others for any reason.
In the diagram below there are the main components of the system, subsystem interconnections and external interfaces to help you understand the main idea of KeePass. All of them are analyzed with more details in this document.



2.2 Product Features

KeePass Password Safe provides the user with the following functions:

- **Database – New, Open, Close, Save, Print, Search, Import, Export**

  User can create a new database locked by a Master Key. The database can be opened and closed whenever user wants it. Changes on the data are permitted and the changes can be saved. The user also can print all data in order to keep them with him even when a computer is not available. Also the user can search the database using key words through a search engine provided with the software. Last but not least, the database can be imported and/or exported from/to the Internet.

- **Group/Subgroup – Add, Modify, Delete, Find**

  Data are organized in groups and subgroups in the order that user wants and finds effective. Those groups can be modified whenever. New groups and subgroups can be added easily and can be deleted the same way. The feature of searching can be applied in just one group and not in the whole database if wanted.

- **Entry – Add, View/Edit, Duplicate, Delete**

  A new entry can be added in any group or subgroup and it contains title, username, password, URL and notes. Not all fields are required for an entry. An entry can be duplicated and deleted in the click of a button.

- **Change Language**

  At KeePass website there are available language translations that can be downloaded and applied easily.

- **Auto-Type**

  The user can select a sequence of keypresses that KeePass will be able to perform and send them to any window or browser.

- **Command Line Options**

  The user can pass a file path in the command line in order for KeePass to open this file at startup.

- **Composite Master Key**

  To open a database you must use all key sources such as password, key file and/or Windows account details that were used when the Master Key was created. All these together form the Composite Master Key and are all required in order to open the database. So the user cannot use a combination of them to unlock the database.

- **Configuration**

  This feature is used to explain how KeePass store its configuration and where.

- **Import/Export**

  KeePass can support importing data from CSV files, Code Wallet, Password Safe and Personal Vault.

- **Integration**

  KeePass uses Global Hot Key to restore KeePass main window and Limit to single instance option to run only one instance of KeePass at a time.

- **Password Generator**

  There are available generations based on character sets and based on patterns the first for generating random passwords and the second for creating passwords which require specific patterns. There is also available generating passwords that follow rules which are determined further down on this document. Then there are security-reducing options which reduce the security of the passwords they are applied to. Finally there are configuring settings of automatically generated passwords for new entries so that a random password will automatically be created by KeePass when a new entry is wanted.

- **Secure Edit Controls**

  KeePass offers the ability for passwords and data to be appeared behind asterisks when the user wants it. When this option is turned on, secure edit controls stronger than the ones of Windows are protecting your data and no one can access them, see them or steal them.

- **TAN Support**

  KeePass uses TAN-Transaction Authentication Numbers for even more security. This feature can be used for generating one time passwords so that there won't be any chance, for anyone to access e.g. your bank account even if he finds out that password. That is because when the password is entered one time it becomes useless. TANs can be added using the TANs wizard.

- **URL Field**

  The URL field supports various special protocols and placeholders and can be used for Standard capabilities where URL field can execute valid URLs for which a protocol handler is defined. In addition to that, KeePass supports all registered protocols that Internet Explorer supports. URL field also offers the ability of executed command lines instead of URLs. Also, placeholders can be used that will be automatically replaced when the URL is executed.

- **Using Stored Passwords**

  Passwords that are stored in the database can be copied to website accounts and applications with security and without retyping them again. This can be done by several methods such us Context-Sensitive Password List, Drag and Drop, Auto-Type and KeeForm. All of them are explained better further down.

- **Lock Workspace**

  Last but not least at all is the locking workspace feature. This feature is turned on and locks the database when minimized. So to unlock it the Master Key is required again. The workspace can be locked manually as well by selecting this option from File menu.

2.3 User Classes and Characteristics

- **Advanced end users:** users that are familiar with programming and can personalize their database by creating auto-types, using command line options and generally can use features and maybe expand their use by adding more functions.

- **End users/Desktop:** users with no particular knowledge on computer programming. They just use the database for organizing their data and to keep them safe.

- **System administrators:** administrators working on computers that support a lot of accounts and personal data for other users. Using KeePass the administrator can save all data with no risk of leak to third persons.

- **Science/Research Telecommunications:** for organizing data that have to do with lots of people and applications

- **Industry:** for one-time passwords that can be used for testing controls or for expired entries to gain access in particular systems and programs.

- **Other Audience**

2.4 Operating Environment

KeePass should run on Operating Systems: WINE, 32-bit MS Windows (95/98), 32-bit MS Windows (NT/2000/XP), All 32-bit MS Windows (95/98/NT/2000/XP),Win2K, WinXP, Microsoft Windows Server 2003.

The user interfaces used are: NET/Mono, Win32 (MS Windows)

All new releases contain

Filename	Architecture	Type
KeePass-1.x-Setup.exe	i386	.exe (32-bit Windows)
KeePass-1.x-Src.zip	Platform-Independent	Source .zip
KeePass-1.x.zip	i386	.zip

and release notes witch describe what has changed and what has been added.

Nothing more than these is required for a fully functional KeePass.

KeePass should run perfectly on older releases without any features limitations or data loss.

2.5 Design and Implementation Constraints

Timing requirements in KeePass Password Safe:

When a password is copied for any reason, (e.g. copy to an application, account, and website) it remains in the memory for only 10 seconds. After 10 seconds pass there is nothing to paste and you have to recopy again. That provides security in a case a password is copied and not pasted anywhere so no one can find it out by pasting later.

Language Requirements in KeePass Password Safe:

Not in all translations translated help files and tutorials are available.

Specific Technologies used in KeePass Password Safe:

- In order to keep the user's data fully protected, 2 very secure algorithms are used:

Cipher	**Block Size**	**Key Size**
Advanced Encryption Standard (AES / Rijndael)	128 bits	256 bits
Twofish	128 bits	256 bits

  In both algorithms every time the user saves a database, a random 128-bit initialization vector is generated.

- For the creation of the 256-bit key the Cipher uses, the Secure Hash Algorithm SHA-256 is used.
- All the bytes needed for the Initialization Vector, the master key salt, etc are generated via pseudo-random sources: current tick count, performance counter, system date/time, mouse cursor position, memory status, active window focus handles, window message stack, process heap status, process startup information and several system information structures.
- When the KeePass is active, all passwords are stored encrypted in process memory so in order for them to be completely safe the ARC4 encryption algorithm is used, using a random 12 bytes long key.

2.6 User Documentation

By downloading KeePass Password Safe, the user also gets:

- A compiled HTML Help file with a tutorial and full help on all features provided
- A KeePass Internet shortcut which take the user in the system's official website where are available downloads, translations, plug-ins and extensions.

3. System Features

System features are organized by use cases and functional hierarchy so that the main functions of the system will be understandable.

3.1 New Database

This feature provides the ability to create a new database

3.1.1 Description

It is the first thing a user must do to begin using KeePass. Its main function is the determination of the master password that will unlock the database from now on

3.1.2 Stimulus/Response Sequences

<u>Data Flow</u>

3.1.2.1 **Basic Data Flow**

1. User opens KeePass and select New->Database
2. User writes his private Master Password and/or selects Key File
3. User selects OK
4. Master Password confirmation: the user retypes Master Password
5. The main database window opens

3.1.2.2 **Alternative Data Flows**

3.1.2.2.1 Alternative Data Flow 1

3. User selects Help
4. The help file opens

3.1.2.2.2 Alternative Data Flow 2

3. User selects Cancel
4. Exit from KeePass

3.1.2.2.3 Alternative Data Flow 3

2a. The user does not determines a Master Password
2b. A message is appeared which prompts him to enter a password or key file

3.1.3 Functional Requirements

REQ-1: KeePass must be downloaded and installed

REQ-2: Master Password has no limits in length. A whole sentence can be used with more than 100 characters.

3.2 Open Database

This feature allows the user to open an existing database.

3.2.1 Description

When choosing to open a database a user is transferred to his documents where he navigates to find the database he wants. When the database is found, the master password is wanted so that the database will be unlocked. Once this is done the user is free to access his data.

3.2.2. Stimulus/Response Sequences

Data Flow

3.2.2.1 **Basic Data Flow**

1. User opens KeePass and select Open->Database
2. User navigates through his folders
3. User selects a database
4. User types Master Password
5. The main database window opens

3.2.2.2 **Alternative Data Flows**

3.2.2.2.1 Alternative Data Flow 1

3a. User selects a type of folder non suitable for database
3b. A message “file not found” appears
3c. User selects another folder

3.2.2.2.2 Alternative Data Flow 2

4a. Master Password is wrong
4b. A message “invalid/wrong key” appears
4c. User types another master key

3.2.2.2.3 Alternative Data Flow 3

3. User chooses cancel
4. Exit from KeePass

3.2.3 Functional Requirements

REQ-3: Folder selected must be of type the database can read and that is “name”.kdb

3.3 Save Database

This feature allows the user to save any changes or updates he has performed to his database.

3.3.1 Description

When a database is opened, the user can access his passwords, organize them into new groups and subgroups, delete and add entries and so much more. But when it is time for the database to close or during his working on the database, he can save the changes made.

3.3.2. Stimulus/Response Sequences

Data Flow

3.3.2.1 **Basic Data Flow**

1. User opens KeePass and changes his data
2. User selects save database
3. Database is saved
4. User exits KeePass

3.3.2.2 **Alternative Data Flows**

3.3.2.2.1 Alternative Data Flow 1

2a. User selects save as
2b. User gives a new database name
2c. New database is saved and opens with the same master password

3.3.2.2.2 Alternative Data Flow 2

4. User continues working after he saves the database

3.3.2.2.3 Alternative Data Flow 3

2a. User wants to exit KeePass
2b. A message is appeared asking if he wants to save the database
2c. User selects yes and exits, or no and exits or cancel and return to database

3.3.2.2.4 Alternative Data Flow 4

2a.Users minimizes the database
2b. A message is appeared asking if he wants to save the database before locking

3.3.3 Functional Requirements

REQ-4: Databases must have different names or else the previews one will be replace if selected

3.4 Print Database

This feature allows user to print a selection of data that are stored in the database.

3.4.1 Description

While working on the database, the user has the option to print data from his database. This can be done by selecting print. When this happens, a list of data types that can be printed are shown and the user can select the data to be printed. More specifically fields that can be selected for printing are: Backup entries, which contain entries in the back up group, password groups, group tree, title, username, password, URL, notes, creation time, last access, last modification, expires, icon, UUID and attachment.

3.4.2. Stimulus/Response Sequences

Data Flow

3.4.2.1 **Basic Data Flow**

1. User opens KeePass
2. User selects print from file menu
3. The list of options opens with checked the fields: password groups, title, user name, password, URL, notes
4. User selects OK
5. Data are print
6. User returns on the main window

3.4.2.2 **Alternative Data Flows**

3.4.2.2.1 Alternative Data Flow 1

3a. User selects some more fields and/or unselects some others.

3.4.2.2.2 Alternative Data Flow 2

3a. User unselects all fields
3b. An empty report is printed

3.4.2.2.3 Alternative Data Flow 3

4. User selects Cancel
5. User returns on the main window

3.4.3 Functional Requirements

REQ-5: There must be entries in the database in order for them to be printed

3.5 Search Database

This feature allows user to search for keywords in his database.

3.5.1 Description

There is the ability to search in the database for usernames, groups, passwords, URLs, notes and titles. This is very useful when the user needs to find out very quickly which password is required in one account or what username he has put on another account. It is not necessary to write in the search field all characters. By writing just one character the database will present all data which contains it or are related with it.

3.5.2. Stimulus/Response Sequences

Data Flow

3.5.2.1 Basic Data Flow

1. User opens KeePass
2. User types a password, user name, URL, word of notes, title or group that exist in the database
3. The list of data related to search word are appeared in the main window

3.5.2.2 Alternative Data Flows

3.5.2.2.1 Alternative Data Flow 1

2. User types two or more words in the search field
3. Nothing appears in the main window

3.5.2.2.2 Alternative Data Flow 2

2. User does not type anything
3. Nothing appears in the main window

3.5.2.2.3 Alternative Data Flow 3

2a. User types part or even just one character of password, user name, URL, word of notes, title or group

3.5.2.2.4 Alternative Data Flow 4

2. User types data not related with the database
3. Nothing appears in the main window

3.5.3 Functional Requirements

REQ-6: All data related to the word must be shown. For example if user types “abc” and abc is part of a password and of a username, both entries must be shown

3.6 Add Group/Subgroup

This feature is used to keep data organized in categories for easier access.

3.6.1 Description

There is the ability to organize data into groups and subgroups. The user can create a new group or subgroups into an existing group. When creating a group/subgroup the user must select a name and then he can add entries into it.

3.6.2. Stimulus/Response Sequences

<u>Data Flow</u>

3.6.2.1 **Basic Data Flow**

1. User opens KeePass and selects to add new group/subgroup
2. User types a name and has the option to select an image
3. User selects OK
4. The group/subgroup is created
5. The access returns to the database main window

3.6.2.2 **Alternative Data Flows**

3.6.2.2.1 Alternative Data Flow 1

2. User leaves the group/subgroup name field empty
3. A message is appeared “Add a name for the group/subgroup”

3.6.2.2.2 Alternative Data Flow 2

3. User selects Cancel
4. The access returns to main window

3.6.3 Functional Requirements

REQ-7: A name is required in order for the new group/subgroup to be created

REQ-8: A subgroup cannot be created when no group is selected

3.7 Modify group/subgroup

This feature allows user to change a name given to a group/subgroup.

3.7.1 Description

When user wants to change a group or subgroup's name he has the ability to do that by modifying it.

3.7.2. Stimulus/Response Sequences

<u>Data Flow</u>

3.7.2.1 **Basic Data Flow**

1. User opens KeePass and selects modify group/subgroup
2. User types the new name and has the option to select an image
3. User selects OK
4. The group/subgroup changes name
5. The access returns to the database main window

3.7.2.2 **Alternative Data Flows**

3.7.2.2.1 Alternative Data Flow 1

2. User leaves the group/subgroup name field empty
3. A message is appeared "Add a name for the group/subgroup"

3.7.2.2.2 Alternative Data Flow 2

3. User selects Cancel
4. The group/subgroup's name remains the same
5. The access returns to main window

3.7.3 Functional Requirements

REQ-9: A name is required in order for the group/subgroup to be renamed

3.8 Delete Group/Subgroup

This feature allows the user to delete a group/subgroup.

3.8.1 Description

When a user wants to delete an existing group/subgroup he can do this by selecting Delete group/subgroup from the edit menu. In order for that to happen, he must have chosen first a group/subgroup.

3.8.2. Stimulus/Response Sequences

Data Flow

3.8.2.1 **Basic Data Flow**

1. User opens KeePass and selects a group/subgroup
2. User selects to delete the group/subgroup
3. A message is appeared to confirm the delete
4. User selects yes
5. The group/subgroup is deleted
6. The main window opens

3.8.2.2 **Alternative Data Flows**

3.8.2.2.1 Alternative Data Flow 1

4. User selects no
5. The group/subgroup is not deleted

3.8.3 Functional Requirements

Not exists.

3.9 Find Group/Subgroup

This feature allows user to find data into a specific group/subgroup.

3.9.1 Description

The user selects a group/subgroup and then he chooses “find in this group” from file menu. Then a window appears which prompts the user to write a sequence of characters (letters and numbers) from one character to 200 (or more). After that a menu of options to choose from appears and the user can select to search for: title, user name, password, URL, notes, group name. Search results appear in the main window.

3.9.2. Stimulus/Response Sequences

Data Flow

3.9.2.1 Basic Data Flow

1. User opens KeePass and selects to find something in a selected group/subgroup
2. User types the word to search for
3. User selects the fields which will be searched
4. User selects OK
5. Results appears at the database main window

3.9.2.2 Alternative Data Flows

3.9.2.2.1 Alternative Data Flow 1

2a. User leaves the find field empty

5. All data from the group/subgroup appear

3.9.2.2.2 Alternative Data Flow 2

3. User leaves all search in boxes unchecked

5. Nothing appears in the main window

3.9.2.2.3 Alternative Data Flow 3

4. User selects cancel

5. The main window appears

3.9.2.2.4 Alternative Data Flow 4

2. User leaves the find field empty

3. User leaves all search in boxes unchecked

5. Nothing appears in the main window

3.9.3 Functional Requirements

REQ-10: In order to find something in a group a word must be placed in the find field

3.10 Add Entry

This feature adds a new entry on the database.

3.10.1 Description

The user can add a new entry on the database by clicking add entry on the main menu. When he does this, a window opens which is actually a form. In this form the user completes the fields he wants. He can fill in fields like: group where the entry will be added, title, user name, password, repeat password, URL, notes, expiration date and attachment file. Not all fields are required for an entry to be created. In fact an entry can be added with no fields at all. By selecting OK the entry is created in the group selected.

3.10.2. Stimulus/Response Sequences

Data Flow

3.10.2.1 **Basic Data Flow**

1. User opens KeePass and selects to add an entry
2. User fills in the form, as many fields as he wants, from 1 to 9
3. User selects OK
5. The new entry is added in the selected group

3.10.2.2 **Alternative Data Flows**

3.10.2.2.1 Alternative Data Flow 1

2a. User fills in the password field
2b. The repeat password field is not identical with the password field
2c. A message appears “the repeated password must be identical with the password

3.10.2.2.2 Alternative Data Flow 2

2a. User leaves all fields empty

3.10.2.2.3 Alternative Data Flow 3

3. User selects help
4. The help file or URL field features or Autotype is opened

3.10.2.2.4 Alternative Data Flow 4

3. User selects cancel
4. The main window opens

3.10.3 Functional Requirements

REQ-11: An entry must belong to a group to be created
REQ-12: When the password field is completed the repeat password field must be completed
REQ-13: Password field and repeat password field must be identical

3.11 View/Edit Entry

This feature allows the user to change or modify an existing entry.

3.11.1 Description

The user can modify the context of an entry he already has added. This can be done by selecting view/edit entry. In order for that to happen, the user must select an entry first. When he does this, the form he filled in when he created or last modified the entry open. Then the user can change the group, title, user name, password, URL, notes, expiration date and attachment file. His changes are saved by clicking OK.

3.11.2. Stimulus/Response Sequences

Data Flow

3.11.2.1 **Basic Data Flow**

1. User opens KeePass and selects to view/edit an existing entry
2. User changes the fields he wants in the form
3. User selects OK
5. The modified entry is saved in the selected group

3.11.2.2 **Alternative Data Flows**

3.11.2.2.1 Alternative Data Flow 1

2a. User changes the password field
2b. The repeat password field is not identical with the password field
2c. A message appears “the repeated password must be identical with the password

3.11.2.2.2 Alternative Data Flow 2

3. User selects help
4. The help file or URL field features or Autotype is opened

3.11.2.2.3 Alternative Data Flow 3

3. User selects cancel
4. The main window opens and the selected entry is not modified

3.11.3 Functional Requirements

REQ-14: An entry must be selected to be viewed or modified

REQ-15: When the password field is changed the repeat password field must be Changed and be identical with password field

3.12 Duplicate Entry

This feature creates an exact copy of the selected entry in the same group.

3.12.1 Description

The user can create an exact copy of an entry. This can be done by selecting an entry and clicking duplicate entry. When this is done the new entry is added in the same group were the first was.

3.12.2. Stimulus/Response Sequences

<u>Data Flow</u>

3.12.2.1 Basic Data Flow

1. User opens KeePass and selects an existing entry
2. User duplicates entry
3. The new entry is added below the first one

3.12.2.2 Alternative Data Flows

Not exist

3.12.3 Functional Requirements

REQ-16: An entry must be selected before it is duplicated

3.13 Delete Entry

This feature allows the user to delete an existing entry

3.13.1 Description

The user can delete an existing entry by selecting it and then by clicking delete entry. When this is done a message appears and informs the user that if he deletes the entry he won’t be able to restore it back.

3.13.2. Stimulus/Response Sequences

<u>Data Flow</u>

3.13.2.1 Basic Data Flow

1. User opens KeePass and selects to delete an existing entry

2. A message appears "Are you sure you want to delete the selected entry?"
3. User selects yes
4. The deleted entry is permanently removed from the database

3.13.2.2 **Alternative Data Flows**

3.13.2.2.1 Alternative Data Flow 1

1a. User selects more than one entry

3.13.2.2.2 Alternative Data Flow 2

3a. User selects no
4a. No entry is deleted

3.13.3 Functional Requirements

REQ-17: An entry must be selected to be deleted

3.14 Change Language

This feature allows user to choose one of the language translations that are available in KeePass

3.14.1 Description

The user can select between a number of translations that are available at KeePass website. There are available translations in Arabic, Brazilian, Bulgarian, Catalan, Chinese (Simplified and Traditional), Czech, Danish, Dutch, English, Estonian, Finnish, French, Galician, German, Greek, Hebrew, Hungarian, Italian, Japanese, Korean, Lithuanian, Macedonian, Malay, Norwegian, Polish, Portuguese, Romanian, Russian, Serbian, Slovak, Spanish, Swedish, Turkish and Ukrainian. All the user has to do is to select change language from the view menu and then the language he wants. If he already has download languages packets he can choose one of them. If he wants another language pack he can select get more language. Doing this he will be transferred at KeePass website and choose the language he wants for downloading.

3.14.2. Stimulus/Response Sequences

Data Flow

3.14.2.1 **Basic Data Flow**

1. User opens KeePass and selects to change language from the view menu
2. The user selects a language to load from the list that appears
3. A message appears that informs user that the installation has been done and in order for the changes to take place KeePass must be restarted

4. User selects yes (restart now)
5. KeePass asks to save any changes that may have been made in the database and restarts
6. User unlocks the database using the Master Password
7. The database appears translated in the new language

3.14.2.2 Alternative Data Flows

3.14.2.2.1 Alternative Data Flow 1

2. User selects get more languages
3. The browser opens in KeePass website were translations are located
4. User selects a language he wants
5. The language file downloads and can be found at the list with available languages in the language menu

3.14.2.2.2 Alternative Data Flow 2

4. User selects no (restart later)
5. No changes are appeared in the database until the next time KeePass restarts

3.14.2.2.3 Alternative Data Flow 3

2. User selects close
3. No changes are made in the database

3.14.3 Functional Requirements

REQ-18: An internet connection and a browser are required in order to download new language translations

For the following features, use cases are not included

3.15 Auto-type

This feature allows user to define a sequence of keypresses which KeePass will automatically perform.

3.15.1 Description

The user defines the sequences which can be sent to any other open window like browsers or login accounts. By default the sent keystroke sequence is {USERNAME}{TAB}{PASSWORD}{ENTER}. There is the ability for the user to define his own auto-type sequence in the notes field of each entry. The sequence must be prefixed with “Auto-Type” and length of one line. If a sequence is more than a line it cannot be used and in case of two sequences in one note’s field only the first one is used. There are 3 methods to invoke auto-type.

- The first method is to use the context menu command and perform auto-type while the entry is selected.
- The second method is to select an entry and press Ctrl+V
- The third method is to use the system-wide auto-type hot key so that KeePass can search all entries in the database for matching sequences.

3.15.2 Functional Requirements

REQ-19: The prefix “Auto-Type:” is required in front of each sequence
REQ-20: Sequence’s length must not be more than a line (59 characters)
REQ-21: If two auto-types are referred in one note field, only the first is used

3.16 Command Line Options

This feature allows user to pass a file patch in the command line so that KeePass will open it immediately after start up.

3.16.1 Description

The database file location is passed as argument in the command line. More about this feature and examples for using it can be found at help contents in KeePass

3.16.2 Functional Requirements

REQ-22: Only one database file is allowed in command line options
REQ-23: In case a space is found in the path, it must be enclosed into quotes

3.17 Composite Master Key

This feature is a composition of master password and key files and all its composites are required so that the database can be unlocked.

3.17.1 Description

The database can be unlocked with a master password, a key file or both of them. But the same composition of them must be used always.
A master key is a password the user creates. Once a master key is created for a database, it is always required. In case the user forgets it, the database can never be opened again by any way.
A key file is a file that locks the database. The database opens when this file is present. If the file is lost and there are no copies of it, all data are gone forever and database never unlocks again.

3.17.2 Functional Requirements

REQ-24: If a master password is required to unlock the database, the database doesn't open unless the password is entered
REQ-25: If a key file is required to unlock the database, the database doesn't open unless The key file is present
REQ-26: If there is a composite key, both master password and key file are required
REQ-27: In case of lost master password or key file, the database never unlocks again. There is no recovery
REQ-28: There is no backdoor or key that unlocks all databases

3.18 Import/Export

This feature gives the ability to user to import/export files from/to database.

3.18.1 Description

There is the ability to import data from CSV files, code wallet, password safe and password vault. There are available plug-ins which add more import capabilities and formats and those are: File format CSV and File format XML. There isn't any standard password database format and every password manager uses its own file format. Despite that, almost all support exporting to CSV or XML files.

3.18.2 Functional Requirements

REQ-29: File formats are not specialized password database formats
REQ-30: File formats only specify a low-level layout of stored data

3.19 Integration

This feature allows switching back from an application to KeePass.

3.19.1 Description

The global hot key takes the user back from one window to KeePass. In case of multiple databases running of KeePass the global hot key restores the window which was opened first of all. The global hot key is Ctrl+Alt+K

3.19.2 Functional Requirements

REQ-31: Global hot key cannot be changed

3.20 Password Generator

This feature generates random passwords.

3.20.1 Description

The password generator creates random passwords every time an entry is created. This password may contain letters (big and small) and numbers. The user can keep it or he can put his own in the field. The generation can be based on character sets, patterns or can be created according by rules. This feature can be disabled if user wants it by selecting 0 as password length in the password generator dialog. More about characters sets and patterns can be found at KeePass Help Files including directions and examples.

3.20.2 Functional Requirements

According to restriction rules that are applied every time.

3.21 TAN Support

This feature allows creation of Transaction Authentication Numbers.

3.21.1 Description

TANs can be passwords that provide security because they can never be used for more than once time. They can be added to user’s database by using TAN wizard and can contain all letters and numbers. When TAN are created, they appear as typical entries which in the field title contain “<TAN>” so that KeePass will know it’s a TAN. In TAN entries, user cannot change the title, user name or URL but notes can be added. When a TAN is used, its expiration time is set to that current time and the entry expires automatically.

3.21.2 Functional Requirements

REQ-32: Title, username or URL cannot be changed in a TAN entry.
REQ-33: When a TAN is used, it expires automatically and can never be used again.

4. External Interface Requirements

4.1 User Interfaces

User interface includes various forms and windows. The main database window consists of the main menu bar with file, edit, view, tools and help. Under main menu there is a toolbar with shortcuts to most used functions of KeePass. Those are: new, open, save, add entry, edit/view entry, delete entry, copy username to clipboard, copy password to clipboard, find in database and lock workspace. On the main database window are appeared entries from a selected group. Groups and subgroups can be found at a side bar.

When a function is performed like adding, editing or deleting, the active window is the one performing the action. At this time the main database window is inactive and cannot be accessed unless the current active window is closed.

KeePass as referred previously uses NET/Mono and Win32 (MS Windows) interfaces.

4.2 Communications Interfaces

- Internet connection and a browser are required in order for several functions to be executed such as downloading plug-ins

5. Other Nonfunctional Requirements

5.1 Performance Requirements

When a password is copied, it remains on memory for only 10 seconds. If in the meanwhile it is not pasted anywhere, it must be copied again. That happens so that if user copies a password and not paste it anywhere, the password cannot be found by anyone later.

5.2 Safety Requirements

When a USB which contains the database is removed from a computer while changes haven’t been completely saved, the database is damaged and cannot be opened. In this case the repair functionality can help by repairing KeePass database file from tools menu.

In case the user forgets or loses the Master Password, the repair functionality won’t help.

In case the header of the database, which is the first few bytes, is corrupted, again the repair functionality won’t help.

To avoid this kind of situations, backups can be done regularly.

5.3 Software Quality Attributes

- KeePass is a small and light project so it does not need to be installed. All it takes is unpacking from the Zip package. It can be transferred also in a USB stick with no additional configuration needed.
- KeePass is a project that once uninstalled from a computer, leaves no trace behind. So there is no way passwords and other data in the database to be found later.
- KeePass is developed under GNU General Public License version 2 or later (copy from http://www.gnu.org/licenses/gpl.html) and can be distributed under those terms.
- KeePass can be found and downloaded from www.sourceforge.com and the project's official website at http://keepass.info/ for free.